Vision and Tracking in a Smart AI Kitchen for Older Adults

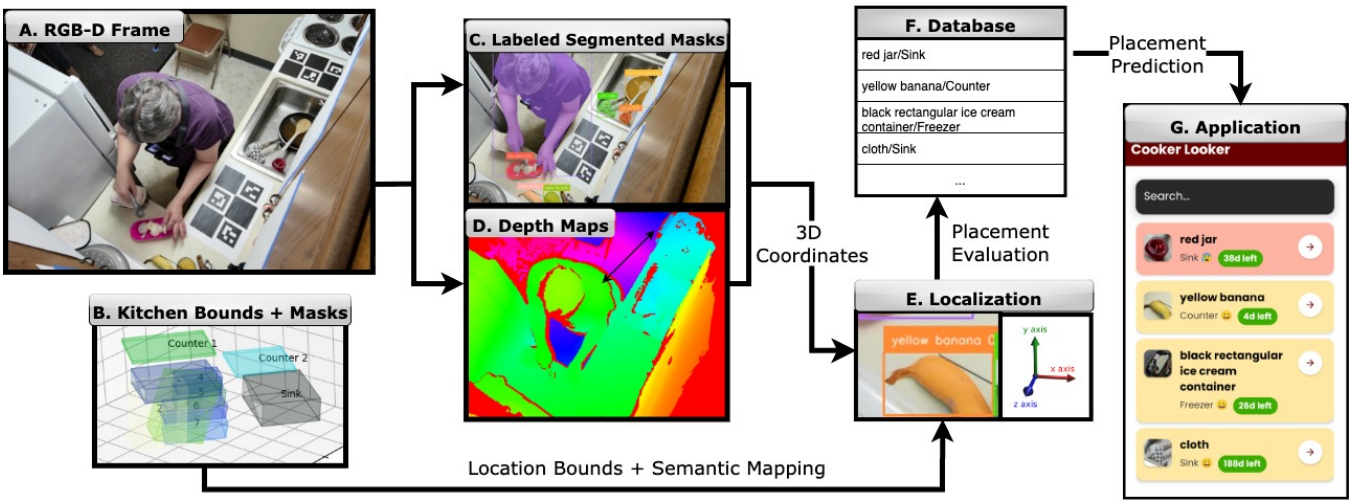

Figure 1: An overview of our novel tracking system: RGB-D frames (A) are processed with kitchen registration (B), segmentation masks (C), and depth maps (D) to localize items in 3D space (E). Item placements are stored and evaluated in a database (F) and displayed on the CookerLooker mobile app (G), helping users to find items.

Abstract

Cognitive decline associated with aging can present a myriad of challenges to everyday functions, such as in locating items, which can hinder an individual's ability to prepare meals. In this work, we propose a system that uses an overhead RGB-D camera and zero-shot object detection models to identify and track items in 3D as they are being used in a kitchen. Our pipeline employs several strategies to achieve stable object tracking and object permanence even despite transient occlusions. We further ground the 3D coordinates of each tracked item using their semantic kitchen locations, which we define using a highly scalable registration procedure. We evaluate our system on a set of eight videos of older adults (median age = 78), captured in two different kitchens, and achieve reasonable performance in complex, ecologically valid scenes. Finally, we demonstrate how our pipeline can inform interaction with a user through CookerLooker, a mobile app that helps users easily locate items in their kitchen. To help catch when items are misplaced, the app includes a module that evaluates the appropriateness of item placements, information that could further inform assistive interactions with the user. In integrating our pipeline with CookerLooker, we attempt to address common challenges faced by older adults in the kitchen, empowering them to age independently.

CCS Concepts

• **Computing methodologies** → **Object detection**; **Tracking**; *Spatial and physical reasoning*; *Reasoning about belief and knowledge*; • **Human-centered computing** → *Accessibility technologies*; *Interaction devices*; • **Applied computing** → Consumer health.

Keywords

tracking kitchen items, real-time object detection, meal preparation, scalable registration, interactive kitchen app

1 Introduction

According to the U.S. Census Bureau, the U.S. population aged 65 and older rose by 3.1% to 61.2 million from 2023 to 2024 [24]. As more Americans enter their golden years, a majority of them desire to age in place i.e. to stay in their own homes and independently perform instrumental daily activities of living (iADLs) such as taking medication, keeping appointments or finding things at home [5, 15]. One key aspect of aging in place is the ability to prepare meals in the kitchen, given that 90% of adults aged 50-80 cook dinner at home for at least three days a week [23]. However, cognitive decline due to aging, like memory loss and slower processing speeds, can present challenges in the kitchen, such as locating items for meal preparation [8, 14, 25]. These challenges can turn cooking into a daily struggle for many older adults, especially for those diagnosed with mild cognitive impairment (MCI) [3, 6, 9] . Therefore, everyday challenges faced in the kitchen can greatly diminish older adults' sense of autonomy and hinder their ability to age in place.

To empower older adults to continue cooking and living independently, our research aims to develop a novel tracking system that helps older adults locate items in their kitchens, alleviating common issues they face in their daily cooking activities. Our system tracks items across the entire kitchen space (e.g., fridge, counters, cabinets, sink) and stores the tracking data in an online database to be accessed at a later time. It achieves considerable tracking accuracy on real-world environments, as demonstrated in a system evaluation on real videos of older adults preparing meals in kitchens.

The contributions of our work are as follows:

- We introduce a novel tracking system that detects kitchen items present in a video stream, determines their locations based on their 3D coordinates, and updates the tracking data in an online database.
- We demonstrate the usability and accuracy of our system by evaluating it on real videos of older adults preparing food in kitchens.
- We develop a registration procedure that enables our system to scale to different kitchen configurations.
- We demonstrate human interaction with our system by introducing CookerLooker, a mobile app with an easy-to-use interface for users to search for their desired items in the kitchen using the system's data and item placement evaluation.

2 Related Work

2.1 Pantry Apps

There are various mobile applications on the market, such as KITCHENPAL [1] and Pantry Check [2], that help users manage their food inventory and track expiration dates. A 2020 paper explores the benefits of such applications, specifically through their own app Pocket Pantry [10]. The researchers found that the application was helpful in reducing food wastage, as customers now knew all of the items in their kitchens, and ensured that customers did not forget any items they still needed.

Our application builds on these applications by creating an intuitive and hands-free approach to taking inventory of food. Instead of users having to manually enter the items they have bought, our application will automatically take inventory with our computer vision detection.

2.2 Smart Fridges

Currently on the market, there are a few fridges that have similar AI capabilities to our project. One of which is the latest Samsung Bespoke 4-Door Flex Refrigerator [22], which uses an internal camera in the fridge to identify any kitchen items that are placed in the fridge. Once these items are identified, the "Image to Recipe" feature can be used to produce recipes using the items stored in the fridge. However, there are limitations to this current model, as the fridge model can only identify 33 kitchen items, and cannot identify expiration dates (users have to enter expiration dates manually).

There has been some academic work done in developing smart fridge models that utilize artificial intelligence to identify kitchen items. One team imagines a smart fridge model that uses cameras and weight sensors to identify food to later give users information on nutrition and improve food sustainability [12]. However, there are limitations in the effectiveness of this work, since this fridge was a conceptual model and was not implemented or validated in real-world kitchens. Additionally, smart fridges cannot track the exact location of items once they are moved around the kitchen.

2.3 More Similar Assistive Technologies

In a 2024 paper, researchers focused on understanding the needs of older adults, such as task reminders and navigation, and then conducted interviews, workshops, and iterative prototyping exercises with older adults to develop a SAR that could help these adults live independently [27]. Similarly, another 2021 paper, conducted a video study of older adults cooking independently at home to analyze the complexity of their cooking tasks [13]. The study identifies some design principles for technology to help older adults cook, such as a need for assistive systems to provide guidance and reminders during meal preparation.

Another system, SafeCook, utilizes multiple-object tracking and computer vision to support safety in the kitchen for older adults. Their system works by detecting specific objects in unsafe positions (like an unattended pot on the stove or a knife left on the edge of a counter) [16]. While this system does a good job of identifying these risky situations, it was not tested on lab or real-world participants, and there was no platform for users to interact with their system. Our work aims to bridge these gaps.

2.4 Object Tracking

Currently, there are several computer vision algorithms available that can track multiple objects [19][4]. One of these is MOTR, which is able to consider both the motion of an item and its appearance to effectively track multiple items [26]. However, by itself, it is not able to track items out of view.

There has been research done to apply object tracking to the kitchen. For example, one approach involved training an algorithm to detect changes in the arrangement of a kitchen. The paper aimed to have the robot sense the environment with a camera and detect what objects have been moved. By following the same path, the agent was able to compare the view of the kitchen and select frames for precise comparison [7]. Similar to our system, they used computer vision to track the location of the objects. However, this paper did not consider objects being moved into places completely out of view of the camera. In addition, users are unable to interact with the kitchen while the robot is scanning.

3 Methodology

In this section, we detail the components of our object detection and tracking pipeline. Our system takes in an RGB-D frame and first passes the RGB image through an object detection module, using Grounded SAM2, that identifies kitchen items from natural language prompts and then produces precise segmentation masks. We apply the masks for the detected objects to depth maps of the scene to localize their position in 3D, which we track over-time. Our tracking algorithm maintains object permanence by handling transient occlusions and using motion-based logic to discern whether reappearing objects correspond to previously tracked items or new instances. Finally, we introduce a registration procedure for our algorithm to scale to different kitchen configurations.

3.1 Object Detection

Our system utilizes a ceiling-mounted, RGB-D camera that captures a video stream in a kitchen environment. We leverage zero-shot detection models, in this case Grounding DINO [20], to detect a set of objects that appear in the kitchen scene given a textual prompt. Let n be the number of frames in capture session F, with f_j denoting the j-th RGB frame, and let $P = \{p_1, \ldots, p_k\}$ be the set of k textual

prompts describing expected kitchen items (e.g., *"yellow banana"*, *"pink plate"*, *"silver knife"*). Each frame f_j is passed as input, along with P, to Grounding DINO, which outputs for every prompt p_i, a bounding box b_i and corresponding confidence score s_i if an item is detected.

A set of bounding boxes $B_j = \{b_1, \ldots, b_k\}$ and a set of corresponding labels $S_j = \{s_1, \ldots, s_k\}$ from Grounding DINO is passed directly as an input to SAM 2 [11, 18], which refines the bounding boxes into pixel-level segmentation masks M_j. Together, Grounding DINO and SAM 2 form the Grounded SAM 2 pipeline [21]. Given a frame f_j and prompts $P = \{p_1, \ldots, p_k\}$, the flow of information is as follows:

$$(f_j, P) \xrightarrow{\text{Grounding DINO}} (B_j, S_j) \xrightarrow{\text{SAM 2}} M_j$$

This integration combines the versatility of prompt-based detection with the precision of segmentation.

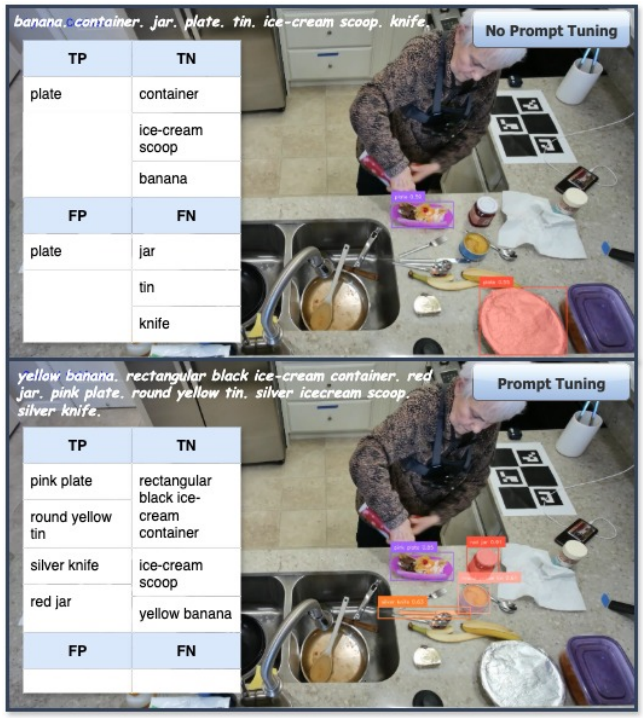

Figure 2: With prompt and threshold tuning, there were fewer false positives, fewer false negatives and higher confidence scores. Without prompt tuning, the plate was correctly identified, but the aluminium foil was also falsely misidentified as a plate (false positive).

3.2 Item Tracking Algorithm

To address gaps left by missed detections and maintain reliable item lists across frames, we introduce a tracking algorithm that infers the continued presence of items even when temporarily undetected by Grounded SAM 2.

The algorithm maintains two dictionaries: `seenCounts`, which records the number of consecutive frames an item has been detected, and `notSeenCounts`, which records the number of consecutive frames it has not been detected. For each frame, if an item is detected, `seenCounts[item]` is incremented and `notSeenCounts[item]` is reset to 0, after which the item is added to the current item list. If an item is not detected, its `notSeenCounts[item]` is incremented. The item is still considered present if: $(\texttt{notSeenCount} < M) \wedge (\texttt{seenCount} >= N)$, where $M = 5$ and $N = 50$ are thresholds on how long an object is unseen vs. seen that impact whether or not an item remains in the item list. Otherwise, the item is removed from the list and its `seenCount` is reset to 0.

This temporal consistency criterion, enforced by thresholds M and N, reduces both false negatives from short occlusions and false positives from misdetections that flicker in and out of view. Increasing N suppresses false detections while increasing M reduces missed detections from longer occlusions. However, if the two thresholds are set too high, new items take too long to register and removed items linger in the list. We selected values for M and N that empirically yielded the best balance between precision and recall.

However, our kitchen-tracking system only works with one camera and cannot detect fully-occluded items in that view. We hope to mitigate this challenge by incorporating more camera views and consolidating them with a 3D world coordinate system. In addition, our system is not robust to changes in item states. For instance, the system will lose track of an apple when it is cut into slices, resulting in a false negative. In the future, we intend to reason about undetected items with activity recognition to overcome more false negatives.

3.3 Achieving Object Permanence

While Grounded SAM 2 enables robust identification, the masks alone are insufficient for tracking items across time and space. To achieve persistent tracking, the system must map these 2D detections into 3D world coordinates. To convert 2D detections into 3D positions, we utilize a localization procedure based on ChArUco calibration boards distributed throughout the kitchen. Using this approach, the set of segmentation masks $M_j = \{m_1, \ldots, m_k\}$ combined with depth data d_j yield 3D item coordinates in the global kitchen frame, providing the spatial foundation for occlusion-aware tracking and object permanence.

While the tracking algorithm described in Section 3.2 smooths over temporary occlusions, it does not by itself guarantee consistent item identity across frames. For example, when an item reappears after being moved, the system must determine whether it is the same instance or a new object. To address this, we implement an object permanence check.

Let $x_j \in \mathbb{R}^3$ denote the 3D position of an item at time j, obtained from localization. For each newly detected item at position x_j, we compare it with the stored position x_{j-k} of previously tracked items of the same class. The instantaneous velocity is estimated as

$$v = \frac{\|x_j - x_{j-k}\|}{\Delta t},$$

where Δt is the elapsed time between frames x_j and x_{j-k}. If $v < v_{\max}$, where $v_{\max} = 50$ is a feasibility threshold based on human

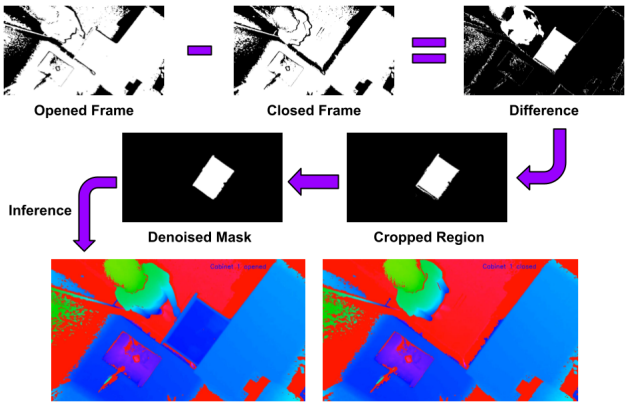

Figure 3: Obtaining the masks of a drawer.

motion constraints in the kitchen, the system considers the detections to represent the same item. Otherwise, the item is treated as a new instance.

This approach has several advantages: It requires no additional setup across different kitchen configurations, and it is invariant to time (an item can be hidden for long periods and still be matched upon reappearance). However, it may introduce false positives in cases where two distinct but similar items move along potential trajectories.

3.4 Kitchen Registration Procedure

Beyond simply localizing the 3D position of objects, the system should also be able to understand where they are semantically in the kitchen environment. To achieve this, we introduce a kitchen registration procedure that extracts the storage masks and 3D bounds of key locations. The registration procedure establishes masks for each storage location (cabinets, drawers, fridge, freezer) and computes 3D bounds for all major kitchen regions (storage, counters, sink). This preprocessing step provides the semantic grounding needed for scalable deployment across different kitchen configurations.

First, storage masks are created by subtracting a depth frame of each cabinet in its open state from a frame of the same cabinet in its closed state, followed by cropping and noise removal (see Figure 3). These masks allow the system to automatically determine whether a cabinet is open or closed by comparing pixel matches against a threshold. Next, the 3D bounds of each location are determined. Each semantic region is modeled as a rectangular prism defined by four hand-selected corner points (see Figure 5). The pixel coordinates of these corners, combined with pose estimation, yield the real-world size, position, and orientation of the location. As shown in Figure 4, the resulting 3D models align closely with the physical kitchen layout.

This procedure is lightweight and requires minimal setup. With a short briefing, non-experts can define location corners, and when a clear corner view is unavailable, dimensions can be manually specified. This method provides a flexible and scalable way to register diverse kitchen environments.

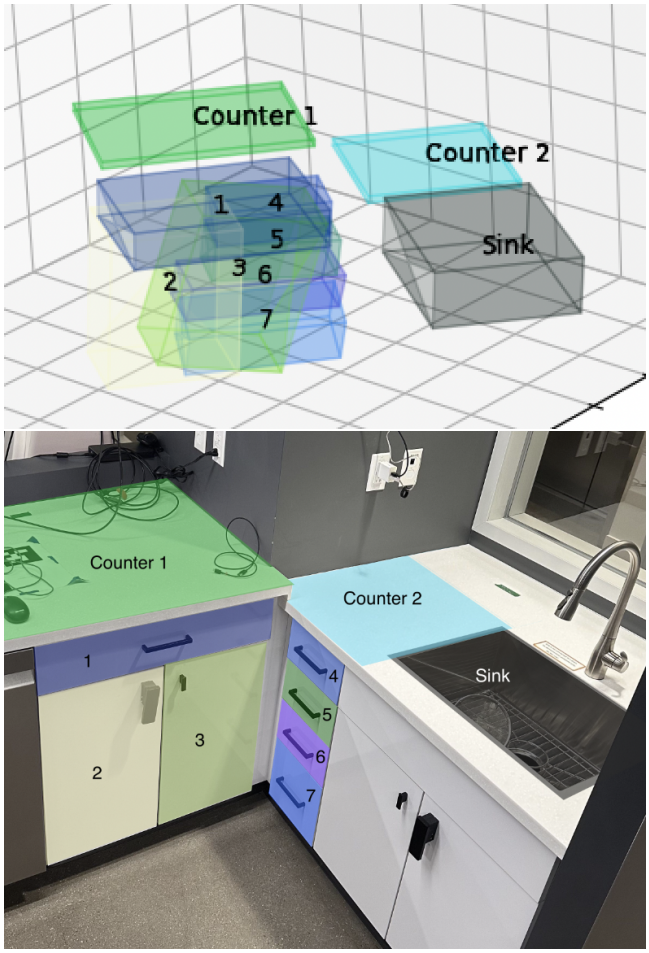

Figure 4: A 3D-plot of various locations in a test kitchen and a labeled picture of those locations in real life. The bounding boxes in the plot represent when the cabinet/drawer is open.

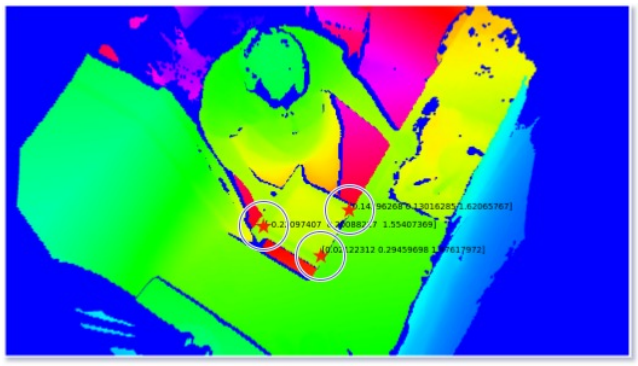

Figure 5: Defining the bounds of the open drawer for registration using 3-4 selected points on the depth map (marked as red stars).

4 Results and Discussion

To evaluate our system, we first ran a study to collect test RGB-D videos in two different model kitchens using the Orbecc Femto Bolt Depth Camera [17]. Each video captured an older adult (median age = 78) preparing a banana split and moving multiple items around the kitchen. At test time, we ran our complete pipeline on eight videos, as well as altered and removed individual system components (detection threshold, prompt tuning, and tracking) as ablations. Table 1 shows the list of all items from the test scene that we attempted to detect and track. Ground truths were obtained by manually labeling the visible items in each frame.

Figure 6: Test kitchens where the study was conducted.

The same registration procedure described in 3.4 was performed on each kitchen configuration before the system was evaluated on the videos. The following ablations/alterations were made to the system:

- **No prompt tuning**: Generic prompts were used in place of descriptive ones to evaluate the effect of prompt tuning. Table 1 shows a list of descriptive and generic prompts for each item.
- **Lower threshold (0.5)**: The Grounding DINO confidence threshold was set to 0.5 to evaluate the effect of using a lower confidence threshold.
- **Higher threshold (0.6)**: The Grounding DINO confidence threshold was set to 0.6 to evaluate the effect of using a higher confidence threshold.
- **No tracking**: The tracking algorithm and existing item check were ablated from the complete system to evaluate their effect on tracking accuracy.

Across all our metrics, we wish to prioritize recall. False negatives in our system mean that users will be unaware of items that are potentially in dangerous positions. With users' safety in mind, we want to ensure that items in the kitchen are consistently tracked, with minimal false negatives.

Besides Lower threshold (0.5) and Higher threshold (0.6), all other test conditions used a Grounding DINO threshold of 0.55, which was determined through threshold tuning on a single test video and was found to yield a high recall while balancing other metrics. All experiments were conducted on a NVIDIA GeForce RTX 3060 Ti GPU.

Thereafter, the performance of the complete system and its ablated/altered counterparts were compared using standard accuracy metrics averaged across the eight videos. In addition, we also considered the mean number of extra instances per item, which reflects how the system sometimes incorrectly treated the same object as more than one instance, due to issues with tracking. Here, fewer extra instances reflect a greater extent of achieving object permanence.

To evaluate the effect of prompt tuning, we compare the complete system against an ablated version where generic prompts (e.g., *"jar"*, *"container"*) were used in place of descriptive ones (e.g., *"red jar"*, *"round yellow tin"*). As shown in Table 2, removing prompt tuning resulted in a large drop in recall (0.34 vs. 0.69) and F1 score (0.45 vs. 0.68), despite achieving slightly higher precision (0.72 vs. 0.68). This indicates that generic prompts caused the detector to miss a significant portion of items (an large increase in false negatives).

For the detection threshold experiment, lowering the threshold to 0.5 improved recall, but reduced precision and increased the mean number of extra instances per item. On the other hand, increasing the threshold to 0.6 improved precision and reduced extra instances, but at the cost of a decrease in recall and overall F1 score. These results demonstrate the tradeoff between sensitivity and reliability when tuning detection thresholds. In cluttered environments, too low a threshold produces mis-detections, while too high a threshold leads to missed items. In our system however, while a balance is optimal, we want to prioritize minimizing false negatives.

From Table 2, the ablated system with no tracking had a decrease in recall and F1 score, showing the effectiveness of the tracking algorithm in reducing false negatives due to temporary occlusions. Moreover, the ablated system with no tracking had a very significant increase in extra items when compared to the systems that included tracking. Without the existing item checks, the system stores new item instances without any previous knowledge of detected items. Thus, the existing item check is essential for distinguishing new and previous instances of an item.

To evaluate the overall effectiveness of our complete system, we organized the top 3 systems with the best performance for each metric in Table 3. Our Complete system performed second best for 3 of the metrics and third best for 1 metric, appearing the second most frequently in the top 3 rankings. Although the Higher threshold system appears in the top 3 rankings for all of the metrics, it has a much lower recall than the Complete system. Hence, the Complete system provides good metric scores overall, while prioritizing a high recall.

5 Research Application: CookerLooker App

To showcase the usability of our tracking system in the real world, we introduce CookerLooker, a mobile app for users to search for desired items in the kitchen. The app was developed with React Native and accesses the system's tracking data from a Firestore database. We designed CookerLooker to be easy-to-use for older adults, with its features explained in the following paragraphs.

Firstly, to enhance reliability and safety, each detected item is first validated by a large language model (Gemini 2.0 Flash) through a prompt template that takes in prior context from items and locations. Then, the LLM assesses whether the predicted location of an item is appropriate given the current kitchen inventory. For example, the model evaluates prompts such as: *Is it appropriate to store 'milk' in a 'cabinet'?* The LLM outputs a ranked (by appropriateness) list of all locations and provides a qualitative risk assessment of the system's predicted location. This step allows the system to flag unusual or unsafe placements, such as perishable items being left

Table 1: Item list and prompts used in our study

Item	Descriptive Prompt	Generic Prompt
Banana	yellow banana	banana
Knife	silver knife	knife
Ice-cream container	rectangular black icecream container	container
Ice-cream scoop	silver icecream scoop	icecream scoop
Plate	pink plate	plate
Cherry jar	red jar	jar
Pineapple tin	round yellow tin	tin

Table 2: Mean Tracking Results across the 8 videos

System	Precision	Recall	Accuracy	F1 Score	Extra Instances/Item	FPS
Complete	0.68 ± 0.12	0.69 ± 0.15	0.66 ± 0.10	0.68 ± 0.14	28.63 ± 36.78	2.60 ± 0.08
No prompt tuning	0.72 ± 0.34	0.34 ± 0.24	0.63 ± 0.12	0.45 ± 0.28	0.74 ± 0.90	3.13 ± 0.30
Lower threshold (0.5)	0.65 ± 0.11	0.74 ± 0.14	0.64 ± 0.09	0.69 ± 0.12	40.92 ± 54.10	2.67 ± 0.08
Higher threshold (0.6)	0.71 ± 0.11	0.58 ± 0.13	0.67 ± 0.09	0.63 ± 0.11	18.05 ± 26.60	2.70 ± 0.18
No tracking	0.69 ± 0.12	0.49 ± 0.11	0.64 ± 0.08	0.57 ± 0.10	796.13 ± 223.06	2.73 ± 0.12

Table 3: Top 3 Systems for each metric

Metric	1st	2nd	3rd
Precision	No prompt tuning	Higher threshold	No tracking
Recall	Lower threshold	**Complete**	Higher threshold
Accuracy	Higher threshold	**Complete**	Lower threshold/No tracking
F1 Score	Lower threshold	**Complete**	Higher threshold
Extra instances/item	No prompt tuning	Higher threshold	**Complete**
FPS	No prompt tuning	No tracking	Higher threshold

out, ensuring that users are not only informed of where items are but also whether they are stored safely.

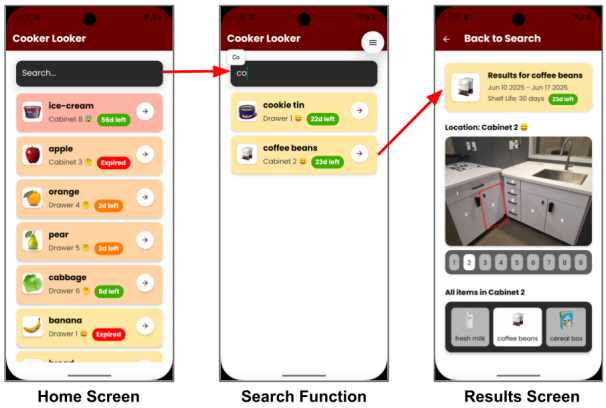

Figure 7: User interfaces of the CookerLooker app.

The results of this evaluation directly inform the CookerLooker interface. The app's Home Screen (see Figure 7) displays a list of all kitchen items with their corresponding locations. The items are sorted and color-coded by the appropriateness of their locations. If an item is in a inappropriate location that requires immediate action, for instance ice cream in a cabinet, it will display a red background and a worried-face emoji, prompting the user to move it to a safer location. If an item is in an inappropriate location but does not require immediate action e.g. an apple in a cabinet, the display has an orange background and a confused-face emoji. Finally, an item in an appropriate location will display a yellow background and a happy-face emoji. The number of days until each item's expiry is also displayed in colored tags (red for expired items, orange for items with 1-3 days left, and green for items with more than 3 days left), after its expiration date is initially entered by the user.

Additionally, CookerLooker has a search feature for users to enter their desired item into the search bar. Clicking on an item pulls up its Results Screen (see Figure 8) with information about when the item was first and most recently seen, shelf life, number of days till expiry, and marked location on the map.

The Results Screen also displays a list of all items in the same location, which users can click on to switch the item in focus. Users can view items in other locations by toggling the buttons just below the kitchen map.

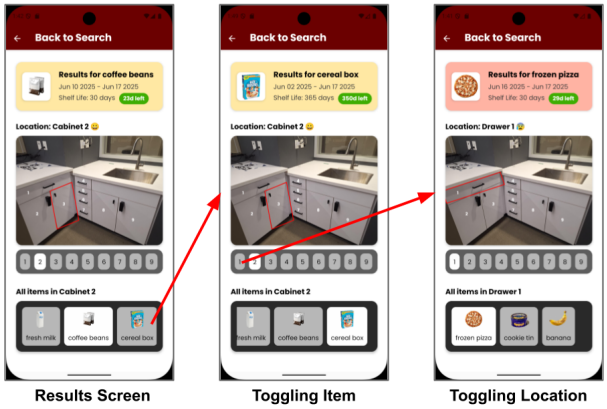

Figure 8: Results Screen of the CookerLooker app.

6 Conclusion and Future Work

In this paper, we introduce a scalable and accurate system for tracking kitchen items in dynamic real-world environments and developed a mobile app that uses the system's tracking data and item placement evaluation to help older adults in daily kitchen activities. Unlike prior approaches that are restricted to single appliances or controlled settings, our pipeline generalizes across different kitchen layouts and directly supports users through CookerLooker, which provides feedback on item storage and helps them search for items in their kitchens.

In the future, we intend to increase tracking accuracy by consolidating multiple cameras and establishing a globalized coordinate system. The data that we collected includes an egocentric view, which can alleviate some of these issues. In addition, we intend to reason about undetected items with activity recognition to overcome more false negatives. We also plan to explore more proactive forms of assistance. For example, detecting when a user appears to be searching for an item, the system could offer assistance to that user. We also want to enrich the system with more precise semantic descriptions of kitchen locations (e.g., "cabinet directly under sink" rather than just "cabinet 4"), allowing users to find items more intuitively. Then, we aim to connect CookerLooker directly to the live tracking system, enabling real-time updates of item locations and notifications when items are misplaced. Through this, we aim to move beyond technical validation toward real-world deployment with users.

Our work contributes to the broader goal of empowering older adults to age in place, reducing caregiver burden and supporting more independent daily living. By making kitchen tasks more manageable, our system moves toward a future of practical assistive technology that preserves autonomy.

References

[1] [n. d.]. KITCHENPAL: Pantry Inventory. https://play.google.com/store/apps/details?id=fr.icuisto.icuisto. Google Play, Accessed: 2025-09-19.

[2] [n. d.]. Pantry check - grocery list. https://apps.apple.com/us/app/pantry-check-grocery-list/id966702368. App Store, Accessed: 2025-09-19.

[3] Alzheimers.gov. [n. d.]. What is Mild Cognitive Impairment? https://www.alzheimers.gov/alzheimers-dementias/mild-cognitive-impairment. Accessed: 2025-09-19.

[4] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. 2016. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE. doi:10.1109/icip.2016.7533003

[5] Joanne Binette and Fanni Farago. 2024. *2024 Home & Community Preferences Among Adults 18 and Older.* Technical Report. AARP Research, Washington, DC. doi:10.26419/res.00831.001

[6] Columbia University Department of Neurology. 2022. One in 10 older Americans has dementia. https://www.neurology.columbia.edu/news/one-10-older-americans-has-dementia. Accessed: 2025-09-19.

[7] Arash Nasr Esfahani, Hamed Hosseini, Mehdi Tale Masouleh, Ahmad Kalhor, and Hedieh Sajedi. 2024. AI-Driven Relocation Tracking in Dynamic Kitchen Environments. In *2024 14th International Conference on Computer and Knowledge Engineering (ICCKE)*. IEEE, 483–488. doi:10.1109/iccke65377.2024.10874520

[8] C. B. Fausset, A. J. Kelly, W. A. Rogers, and A. D. Fisk. 2011. Challenges to Aging in Place: Understanding Home Maintenance Difficulties. *Journal of Housing for the Elderly* 25, 2 (2011), 125–141. doi:10.1080/02763893.2011.571105

[9] K. Jekel, M. Damian, C. Wattmo, L. Hausner, R. Bullock, P. J. Connelly, B. Dubois, M. Eriksdotter, M. Ewers, E. Graessel, M. G. Kramberger, E. Law, P. Mecocci, J. L. Molinuevo, L. Nygård, M. G. Olde-Rikkert, J. M. Orgogozo, F. Pasquier, K. Peres, E. Salmon, and L. Frölich. 2015. Mild cognitive impairment and deficits in instrumental activities of daily living: a systematic review. *Alzheimer's Research & Therapy* 7, 1 (2015), 17. doi:10.1186/s13195-015-0099-0

[10] C. Y. Kim, D. Chou, M. Reininger, and Y. Reiss. 2020. *Pocket Pantry: A Smart Kitchen Storage System.* Technical Report. University of Maryland, College Park.

[11] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. 2023. Segment Anything. *arXiv:2304.02643* (2023).

[12] N. A. Kumar J, S. Suresh, and K. N. Vaani. 2019. An AI driven approach for smart refrigerator to enhance family diet and sustainability. In *Proceedings of the Alliance International Conference on Artificial Intelligence and Machine Learning (AICAAM 2019)*. Bangalore, India.

[13] Sanna Kuoppamäki, Sylvaine Tuncer, Sara Eriksson, and Donald McMillan. 2021. Designing Kitchen Technologies for Ageing in Place: A Video Study of Older Adults' Cooking at Home. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 2, Article 69 (June 2021), 19 pages. doi:10.1145/3463516

[14] D. L. Murman. 2015. The Impact of Age on Cognition. *Seminars in Hearing* 36, 3 (2015), 111–121.

[15] National Institute on Aging. 2023. Aging in Place: Growing Older at Home. https://www.nia.nih.gov/health/aging-place/aging-place-growing-older-home. Accessed: 2025-09-19.

[16] Hubert Ngankam, Philippe Dion, Hélène Pigot, and Sylvain Giroux. 2023. Real-Time Multiple Object Tracking for Safe Cooking Activities. In *Digital Health Transformation, Smart Ageing, and Managing Disability*, Kim Jongbae, Mounir Mokhtari, Hamdi Aloulou, Bessam Abdulrazak, and Lee Seungbok (Eds.). Springer Nature Switzerland, Cham, 192–204.

[17] Orbbec. 2025. Femto Bolt Depth Camera. https://orbbec3d.com/femto-bolt/. Accessed: 2025-09-19.

[18] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. 2024. SAM 2: Segment Anything in Images and Videos. arXiv:2408.00714 [cs.CV] https://arxiv.org/abs/2408.00714

[19] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You Only Look Once: Unified, Real-Time Object Detection. arXiv:1506.02640 [cs.CV] https://arxiv.org/abs/1506.02640

[20] Tianhe Ren, Qing Jiang, Shilong Liu, Zhaoyang Zeng, Wenlong Liu, Han Gao, Hongjie Huang, Zhengyu Ma, Xiaoke Jiang, Yihao Chen, Yuda Xiong, Hao Zhang, Feng Li, Peijun Tang, Kent Yu, and Lei Zhang. 2024. Grounding DINO 1.5: Advance the "Edge" of Open-Set Object Detection. arXiv:2405.10300 [cs.CV]

[21] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. 2024. Grounded SAM: Assembling Open-World Models for Diverse Visual Tasks. arXiv:2401.14159 [cs.CV]

[22] Samsung. [n. d.]. Bespoke refrigerators. https://www.samsung.com/us/home-appliances/bespoke/refrigerators/. Accessed: 2025-09-19.

[23] University of Michigan. 2020. *The Joy of Cooking and its Benefits for Older Adults.* Technical Report. National Poll on Healthy Aging.

[24] U.S. Census Bureau. 2025. Older Adults Outnumber Children in 11 States and Nearly Half of U.S. Counties. https://www.census.gov/newsroom/press-releases/2025/older-adults-outnumber-children.html. Accessed: 2025-09-19.

[25] World Health Organization. 2024. Ageing and health. https://www.who.int/news-room/fact-sheets/detail/ageing-and-health. Accessed: 2025-09-19.

[26] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. 2022. MOTR: End-to-End Multiple-Object Tracking with Transformer. arXiv:2105.03247 [cs.CV] https://arxiv.org/abs/2105.03247

[27] Emily Zhou, Zhonghao Shi, Xiaoyang Qiao, Maja J. Matarić, and Ava K. Bittner. 2021. *Designing a Socially Assistive Robot to Support Older Adults with Low Vision.*

Springer International Publishing, 443–452. doi:10.1007/978-3-030-90525-5_38